PATENT AF
IFW

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of:		:		
	Yoav RAZ, et al.	:		
		:		
Appln. No.:	09/715,681	:	Art Unit:	2135
		:		
Filed:	November 17, 2000	:	Examiner:	DADA, Beemnet W.
		:		
For:	**PHYSICAL SCANNING OF STORAGE BASED APPARATUS FOR ANTIVIRUS**	:	Docket No.:	EMS-00202
		:	**Customer No.**	**52427**

Certificate of Mailing

I hereby certify that the foregoing documents are being deposited with the United States Postal Service as first class mail, postage prepaid, in an envelope addressed to: Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on this date of March 19, 2009.

Name: Anne E. Saturnelli

TRANSMITTAL OF APPEAL BRIEF

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Applicant hereby submits the originally-signed Appeal Brief with Certificate of Mailing, Evidence Appendix and Postcard Receipt. Please charge the amount of $540 for the Appeal Brief to our **Deposit Account No. 05-0889**.

Although we believe that we have appropriately provided for any fees due in connection with this submission, the Commissioner is authorized to credit any overpayment or charge any deficiencies to/from our **Deposit Account No. 503596**.

Should there be any questions after reviewing this paper, the Examiner is invited to contact the undersigned at 508-898-8603.

Respectfully submitted,
MUIRHEAD AND SATURNELLI, LLC

Anne E. Saturnelli
Reg. No. 41,290

March 19, 2009
Date

Muirhead and Saturnelli, LLC
200 Friberg Parkway, Suite 1001
Westborough, MA 01581

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

CERTIFICATE OF MAILING

I hereby certify that the foregoing document is being deposited with the United States Postal Service as first class mail, postage prepaid, "Post Office to Addressee", in an envelope addressed to: Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA, 22313-1450 on March 19, 2009.

Anne E. Saturnelli

* * * * *

APPEAL BRIEF UNDER 37 C.F.R. § 41.37

Application Serial No.: 09/715,681

Filed: November 17, 2000

Applicants/Appellants: Yoav RAZ, et al.

Title: PHYSICAL SCANNING OF STORAGE BASED
APPARATUS FOR ANTIVIRUS

Appeal from a decision of the Primary Examiner dated October 27, 2008

Atty. Docket: EMS-00202

REAL PARTY IN INTEREST

The above-identified application is assigned to EMC Corporation by virtue of an Assignment recorded by the U.S. Patent and Trademark Office on November 17, 2000, at Reel 011309 / Frame 0976.

RELATED APPEALS AND INTERFERENCES

Appellants are not aware of any other appeals or interferences related to the above identified application.

STATUS OF CLAIMS

This is an appeal from a decision of the Primary Examiner in the Office Action dated October 27, 2008 rejecting Claims 1, 3-7, 22, 24-28, 41, 43-52 and 63-71 in the above identified patent application. Claims 1, 3-7, 22, 24-28, 41, 43-52 and 63-71 stand rejected under 35 U.S.C. 103(a). Claims 2, 8-21, 23, 29-40, 42, and 53-62 have been canceled. No claim has been allowed or held allowable. A Notice of Appeal was submitted on January 19, 2009.

STATUS OF AMENDMENTS

In response to a Final Office Action dated June 3, 2008, Appellants filed an after-final Amendment and Response on August 25, 2008 in which claim amendments were made. An Advisory Action dated September 19, 2008 indicated that the Amendment and Response of August 25, 2008 would not be entered because the amendments made therein raised new issues requiring further consideration and/or searching. A Request for Continued

Examination (RCE) with an appropriate petition for extension of time was filed on October 2, 2008 requesting entry of the August 25, 2008 Amendment and Response. A non-final Office Action dated October 27, 2008 was received. A Notice of Appeal was filed on January 19, 2009. Accordingly, all proposed claim amendments have been appropriately entered in the above-captioned application. The claims involved in this Appeal are set forth in the attached Claims Appendix.

SUMMARY OF CLAIMED SUBJECT MATTER

I. Background

A computer system may be attacked by so-called "viruses", which, in many instances, contain code that adversely affects operation of the computer system. Although viruses may exist as stand-alone data files, viruses may also be stored as part of an existing file and are sometimes hidden as seemingly innocuous parts of the file. Thus, a computer system may be infected with a virus by modifying a small portion of a file that is otherwise used for conventional operations unrelated to the virus. When the file is subsequently accessed, the virus may be activated and may cause damage to other parts of the computer system by, for example, replicating itself and/or destroying portions of other files on the computer system.

Antivirus software is provided by a number of commercial vendors to detect viruses on a computer system and, in some instances, remove the offending viruses. Most antivirus software works by scanning individual files to search for suspect patterns of known viruses. Thus, as new viruses are created and detected by the makers of antivirus software, the antivirus software is updated to take into account these new viruses and detect the corresponding patterns. Commercially-available antivirus software may be configured to operate on a single user computer. The antivirus software may run each time the computer is booted up and may scan each file for suspect patterns. However, it may be desirable to run antivirus software for one or more host processors that store and retrieve data using a multihost storage device containing a plurality of host interface units, disk drives, and disk interface units.

One way to perform antivirus checking on a multihost storage device is to run conventional single user antivirus software on each of the hosts so that files of the multihost storage device that belong to each host may be separately scanned by each host. However, such an arrangement may not provide for efficient coordination of the antivirus software for the entire multihost storage device. In addition, if one or more of the hosts do not properly run antivirus software, then viruses may exist on the multihost storage device even though other hosts have performed appropriate antivirus checking. In addition, such an arrangement may be inefficient with respect to updating the data base of known viruses when each of the hosts is separately updated with new virus information.

II. Appellants' Claimed Invention

Appellants' independent claims are discussed below in connection with the specification and Figures for purposes of example and explanation only in accordance with 37 C.F.R. 41.37(c)(v).

Claim 1 recites a computer implemented method of scanning a storage device for viruses, comprising: determining, by the storage device, each track of the storage device that has been accessed for a write operation since a previous virus scan using information about tracks of the storage device without using file-based information, the file-based information including information about file structure, file system, and file type (See, for example, page 15, lines 14-20; the multihost storage device 22 of Figure 4A; page 16, lines 4-12; use of the table 60 of Figure 5; page 17, lines 5-15; page 18, lines 7-17; page 19, lines 1-10); providing, to an antivirus unit by the storage device, information indicating which tracks of the storage device have been accessed for a write operation since the previous virus scan (See, for example, use of the second line 58 of Figure 4A; page 16, lines 4-12 and lines 15-17; use of the table 60 of Figure 5; page 17, lines 5-15; page 18, lines 7-17; page 19, lines 1-10); and scanning, by the antivirus unit using the information provided by the storage device, at least a portion of each track identified as having been accessed for a write operation since the previous virus scan for viruses, wherein scanning is performed without using the file-based information (See, for example, the antivirus unit 26 of Figure 4A; page 16, lines 4-12; use of the table 60 of Figure 5; page 17, lines 5-15; page 18, lines 7-17; page 19, lines 1-10).

Claim 66, which depends from Claim 1, recites wherein the antivirus unit is included in the storage device (See, for example, Figure 6, antivirus units 86, 87, 88 of device 22; page 18, line 18-page 19, line 11).

Claim 67, which depends from Claim 66, recites wherein the antivirus unit is included in a disk controller of the storage device (See, for example, Figure 6, antivirus units

86, 87, 88, respectively, of controllers 76, 77 and 78 in the device 22; page 18, line 18-page 19, line 11).

Claim 68, which depends from Claim 67, recites wherein the antivirus unit is included as software running on the disk controller (See, for example, Figure 6, antivirus units 86, 87, 88, respectively, of controllers 76, 77 and 78 in the device 22; page 18, line 18-page 19, line 11).

Claim 22 recites a computer program product for scanning a storage device for viruses, the computer program product including a computer-readable medium with executable code stored thereon for: determining, by the storage device, each track of the storage device that has been accessed for a write operation since a previous virus scan using information about tracks of the storage device without using file-based information, the file-based information including information about file structure, file system, and file type (See, for example, page 15, lines 14-20; the multihost storage device 22 of Figure 4A; page 16, lines 4-12; use of the table 60 of Figure 5; page 17, lines 5-15; page 18, lines 7-17; page 19, lines 1-10); providing, to an antivirus unit by the storage device, information indicating which tracks of the storage device have been accessed for a write operation since the previous virus scan (See, for example, use of the second line 58 of Figure 4A; page 16, lines 4-12 and lines 15-17; use of the table 60 of Figure 5; page 17, lines 5-15; page 18, lines 7-17; page 19, lines 1-10); and scanning, by the antivirus unit using the information provided by the storage device, at least a portion of each track identified as having been accessed for a write operation since the previous virus scan for viruses, wherein scanning is performed without using the file-based information (See, for example, the antivirus unit 26 of Figure 4A; page 16, lines 4-12; use of the table 60 of Figure 5; page 17, lines 5-15; page 18, lines 4-17; page 19, lines 1-10).

Claim 41 recites an antivirus unit, comprising: means for coupling to at least one storage device (See, for example, 56 and/or 58 of Figure 4A; page 16, lines 4-9 and lines 15-17); means for determining each track of the storage device that has been accessed for a write operation since a previous virus scan using information about tracks of the storage device without using file-based information, the file-based information including information about file structure, file system, and file type (See, for example, page 15, lines 14-20; the antivirus unit 26 and device 22 of Figure 4A; page 16, lines 4-12; use of the table 60 of Figure 5; page 17, lines 5-15; page 18, lines 7-17; page 19, lines 1-10); means for receiving, from the at least one storage device, information determined by the at least one storage device indicating which tracks of the at least one storage device have been accessed for a write operation since the previous virus scan (See, for example, the antivirus unit 26 receiving information from the device 22 over 58 of Figure 4A; page 16, lines 4-12 and lines 15-17; use of the table 60 of Figure 5; page 17, lines 5-15; page 18, lines 7-17; page 19, lines 1-10); and means for scanning, using the information provided by the storage device, at least a portion of each track identified as having been accessed for a write operation since the previous virus scan for viruses, wherein scanning is performed without using the file-based information (See, for example, the antivirus unit 26 of Figure 4A; page 16, lines 4-12; use of the table 60 of Figure 5; page 17, lines 5-15; page 18, lines 7-17; page 19, lines 1-10).

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

I. Claims 1, 3-7, 22, 24-28, 41, 43-52 and 63-66 and 71 stand rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,928,555 to Drew in view of U.S. Patent No. 6,094,731 to Waldin, et al.

II. Claims 67-70 stand rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,928,555 to Drew in view of U.S. Patent No. 6,094,731 to Waldin, et al. and further in view of U.S. Patent No. 6,802,028 to Ruff et al..

ARGUMENT

I. The Examiner has failed to establish a prima-facie case of obviousness under 35 U.S.C. §103(a) of Claims 1, 3-7, 22, 24-28, 41, 43-52 and 63-66 and 71 as being unpatentable over U.S. Patent No. 6,928,555 to Drew in view of U.S. Patent No. 6,094,731 to Waldin, et al.

A. Obviousness Standard

In determining whether or not there is a proper case of obviousness, it is necessary to establish whether one of ordinary skill in the art would, having the prior art references before him, be capable, or otherwise motivated, to make the proposed combination, modification or substitution so as to yield all elements of a claimed invention. *See KSR Int'l Corp. v. Teleflex Inc.*, 127 S. Ct. 1727, 82 USPQ2d 1385 (2007); *see also In re Lintner*, 458 F.2d 1013, 1016 (CCPA, 1972). In rejecting claims under 35 U.S.C. §103, it is incumbent upon the Examiner to establish a factual basis to support the legal conclusion of obviousness and the Examiner is expected to make the factual determinations set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 17, 148 USPQ 459, 467 (1966). *See also United States v. Adams,* 383

U.S. 39 (1966); *Anderson's-Black Rock, Inc. v. Pavement Salvage Co.*, 396 U.S. 57 (1969); and *Sakraida v. AG Pro, Inc.*, 425 U.S. 273 (1976). The analysis used to combine prior art teachings to invalidate a patent claim based on obviousness should be explicitly articulated. *See KSR*, 82 USPQ2d at 1396, citing *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006) ("[R]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness"). However, the analysis may take account of the inferences and creative steps that a person of ordinary skill in the art would employ. Id.

Furthermore, if a proposed modification would render the prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. *See In re Gordon*, 733 F.2d 900, 902, 221 USPQ 1125, 1127 (Fed. Cir. 1984). In addition, if the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims *prima facie* obvious. *See In re Ratti*, 270 F.2d 810, 123 USPQ 349 (CCPA 1959).

B. Claim Interpretation

"[T]he words of a claim 'are generally given their ordinary and customary meaning.'" *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (en banc) (internal citations omitted). The "ordinary and customary meaning of a claim term is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of the patent application." *Id.* at 1313. The "'ordinary meaning' of a claim term is its meaning to the ordinary artisan after reading the entire patent." *Id.* at

1321. The "specification may define claim terms by implication such that the meaning may be found in or ascertained by a reading of the patent documents." *Id.* at 1321.

C. The cited references of U.S. Patent No. 6,928,555 to Drew and U.S. Patent No. 6,094,731 to Waldin, et al. do not disclose or fairly suggest every element of Appellants' claimed invention as to have rendered Appellants' claimed invention obvious to one of ordinary skill in the art at the time the invention was made.

The Examiner rejects Claims 1, 3-7, 22, 24-28, 41, 43-52 and 63-66 and 71 under 35 U.S.C. 103(a) as being obvious over U.S. Patent No. 6,928,555 to Drew (hereinafter referred to as "Drew") in view of U.S. Patent No. 6,094,731 to Waldin, et al. (hereinafter referred to as "Waldin"). Appellants traverse this rejection as set forth below and respectfully request that the rejection be reversed.

In following paragraphs, reference made to an Office Action refers to the non-final Office Action dated October 27, 2008 unless otherwise noted.

The Drew reference discloses a method and apparatus for minimizing file scanning by anti-virus programs. Col. 3, lines 40-55 and Col. 4, lines 5-25 of Drew are cited by the Office Action as support for disclosing determining, by a storage device, each track of the storage device that has been accessed for a write operation since a previous scan using information about tracks of the storage device; providing to an antivirus unit by the storage device, information indicating which tracks of the storage device have been accessed for a write operation since the previous scan; and scanning, by the antivirus unit using the information provided by the storage device, at least a portion of each track identified as having been accessed for a write operation since the previous scan for viruses.

Col. 3, Lines 40-55 of Drew refer to steps of the flowchart of Drew's Figure 2 with respect to processing performed with reference to Figure 1 in which an antivirus program is included in the network server computer 4. After opening a file for write access (step 22), the file is scanned for viruses (step 24). If no viruses are detected, the file is provided to the application program (step 26). A period of time after the file is opened, a file closure request is made (step 28). Upon the file closure request being made, the typical antivirus program loaded into a computer system, such as network server computer 4, interfaces with the network operating system to scan the file for viruses. Col. 4, Lines 5-25 of Drew make reference to Drew's Figure 3 which includes the steps of Figure 2 with new steps 40 and 42. Step 40 determines whether a file was actually written, that is modified by the user performing some writing step on the open file. This latter citation of Drew discloses use of a modification flag set by the operating system. The computer coding for step 40 determines whether an open file was actually written or modified by looking for a flag in the operating system indicative of such a modification. The Office Action states that Drew is silent on the determining step being performed without using file-based information, the file-based information including information about file structure, file system and file type, and performing scanning without using the file-based information.

The Waldin reference discloses a system, method and computer readable medium for examining a file associated with an originating computer to determine whether a virus is present within the file. Waldin discloses scanning and hashing file sectors and placing into a critical sectors file the identification (e.g., number) of each sector that is scanned. (See, for example, Abstract of Waldin). The Office Action cites to Waldin as support for disclosing those features recited in Claim 1 which the Office Action contends are not disclosed in Drew. In particular, the Office Action cites to Waldin as support for disclosing determining physical portions of the storage device that have been modified since a previous virus scan using information about the physical portions without using file-based information, the file-based information including information about file structure, file system and file type (citing to Waldin at Col. 2, Lines 57-64; Col. 6, Lines 37-47; and Col. 3, Lines 5-45); and performing scanning without using the file-based information (citing to Waldin at Col. 6, Lines 43-46; Col. 7, Lines 37-46; Col. 7, Line 64-Col. 8, Line 8; Col. 3, Lines 5-45).

Appellants' independent Claim 1 recites a computer-implemented method of scanning a storage device for viruses comprising determining, by the storage device, each track of the storage device that has been accessed for a write operation since a previous virus scan using information about tracks of the storage device without using file-based information, the file-based information including information about file structure, file system, and file type. Further, Claim 1 also recites scanning, by the antivirus unit using the information provided by the storage device, at least a portion of each track identified as having been accessed for a write operation since the previous virus scan for viruses, wherein scanning is performed without using the file-based information.

Appellants respectfully submit that Drew and Waldin, taken alone or in combination, do not disclose or fairly suggest at least the above-noted features of Claim 1. As pointed out above, the Office Action (see page 4) states that Drew is silent on performing the recited determining and scanning steps without using the recited file-based information and contends that Waldin discloses the recited determining and scanning steps without the recited file-based information. As set forth in detail below, Appellants respectfully submit that, in contrast to what the Office Action contends, Waldin does not disclose or suggest performing the recited determining and scanning steps without using the recited filed-based information including information about file structure, file system and file type.

In addition to the above-noted disclosures in Waldin, Waldin discloses scanning a file and placing into a critical sectors file the identification (e.g., number) of each sector that is scanned. As each sector is operated upon, a hash value is calculated for that sector and inserted into the critical sectors file along with the size of the file scanned. (Col. 4, Lines 52-64; Figures 1 and 2). Waldin's Figure 1 includes antivirus modules on an originating computer 2 and a recipient computer 11 and processing performed on each of the computer systems when transmitting a file from an originating computer to a recipient computer. (See Figure 1; Col. 3, Lines 22-34). Waldin's Figure 3 determines if computed hash values for file 1 match stored hash values for file 1. If not, the entire file 1 is rescanned. (Steps 36, 37 of Figure 3; Col. 6, Lines 43-46; See also Col. 2, Lines 24-26). Waldin discloses determining hash values for only those sectors of a file actually retrieved by module 5 of Figure 1. Module 3 of Waldin's Figure 1 always scans the same set of sectors of a file unless the file changes in length or the contents of those sectors changes in some way. The antivirus accelerator module 5 automatically hashes all sectors scanned by module 3 in the same way regardless of contents of the sectors. No new parser of hasher coding needs to be performed and incorporated into module 5 to support new file formats. (Col. 7, Line 35-Col. 8, Line 2).

The citations of Waldin set forth in the Office Action as support for disclosing features recited in Claim 1 appear to relate to hashing, parsing and scanning techniques that do not require additional programming each time a new virus hosting file format is released (See, for example, Col. 3, Lines 41-43). However, the disclosed techniques of Waldin for hashing, parsing and scanning file sectors appear to require use of at least one of the recited types of file-based information of Claim 1. Waldin discloses operating on files, such as by scanning, and uses information about files. For example, Waldin discloses scanning sectors of a file (see, for example, element 1, Figure 1; step 22 of Figures 2 and 4), and using the size of a file (see step 57, Figure 5) as part of determining when to rescan since a change in file size is an indication that the file contents have changed (See, for example Col. 6, Lines 18-25). In order to operate on files, such as with scanning as disclosed in Waldin, information about the file is used by Waldin. As an example, in order for Waldin to scan a file, the storage locations (e.g., location on a storage device or memory) associated with the file are needed. Without the storage locations as may be obtained, for example, using file system information, Waldin could not even determine what data to scan. In other words, it appears inherent in Waldin's processing that information which is akin to at least file system information is required. Furthermore, Waldin discloses other specific uses of information about a file, such as use of the file size in connection with Waldin's Figure 5 noted above, where the file size is typically one item of information about a file as may be maintained by a file system. It is respectfully submitted that file size is akin to at least information about a file system. Appellants respectfully submit that although Waldin discloses hashing scanned sectors in the same way regardless of contents of the sectors (See, for example, Waldin at Col. 7, Line 35-Col. 8, Line 2), Waldin's techniques still require use of file-based information, such as at least file system information, in order to be used with respect to processing such as scanning disclosed in Waldin.

In connection with accessing files as pointed out above, such as in connection with performing scanning and other processing in Waldin, Appellant respectfully submits that Waldin's techniques require use of file-based information, such as at least information about a file system as recited in Claim 1. Appellants' application, for example, starting at page 19, line 12, supports such an interpretation of file system information as related to file-based information as recited in Claim 1. Page 19, lines 12-15 of Appellants' application indicates that file system information may be provided that allows the antivirus unit to access individual files stored on disk drives. The foregoing file system information may include directory and file system type information. Evidence Appendix PAGE A1 (Page 213 of Microsoft Computer Dictionary, Fifth Edition, which was cited in the list of references by the Examiner accompanying the Office Action dated October 5, 2005) specifies that a file system consists of files, directories or folders and the information needed to locate and access these items. Thus, it is respectfully submitted that one of ordinary skill in the art would understand that file system information as related to the recited file-based information of Claim 1 includes information needed to locate and access the file being scanned in Waldin. Furthermore, it is respectfully submitted that the file size as specifically used in Waldin's processing is also file system information in that information such as file size may be needed as part of processing, for example, to access a stored file to perform scanning in connection with Waldin's described techniques. Additionally, Evidence Appendix PAGE A2 (Page 223 from "UNIX INTERNALS, The New Frontiers" by Uresh Vahalia, which was submitted by Appellants in an Information Disclosure Statement filed December 14, 2005) indicates that a file system maintains a set of attributes for each file. In a UNIX implementation, the attributes may be stored in a structure referred to as an "inode" or index node. Some commonly supported attributes specified in an inode as information maintained by the file system include a file size. Thus, it is respectfully submitted that one of ordinary skill in the art would understand that file system information as related to the recited file-based information of Claim 1 includes the size of file as described in connection with processing in Waldin.

Appellants' Claim 1 also recites in the determining step that the storage device determines tracks accessed for a write operation since a previous virus scan, and, in the recited providing step, that the storage device provides to the antivirus unit information indicating which tracks have been accessed for a write operation since the previous virus scan. As discussed above, pages 3-4 of the Office Action cite to Drew at Col. 3, Lines 40-55 and Col. 4, Lines 5-25 as support for disclosing that the storage device performs the foregoing determining step and that the storage device (in the providing step) provides the recited information to the antivirus unit. Appellant respectfully submits that Drew appears silent regarding the storage device (e.g. storage device 18 and/or cache buffers in Figure 1 of Drew) performing the foregoing features in connection with the determining and providing steps as recited in Claim 1. Based on Appellant's understanding of Drew, Drew discloses use of a modification flag set by the operating system of the server computer. Furthermore, there appears to be no disclosure or suggestion in Drew of the server computer, or the antivirus program thereon, receiving any information from a storage device. Rather, Drew appears to indicate that the server computer does not receive information regarding file modifications from a storage device since the modification flag is set by the operating system. Thus, there appears to be no reason for the storage device to provide any information regarding write operations to the antivirus program on the computer 4 of Drew since such information is provided by the operating system such as prior to writing the file back into memory (e.g., such as file storage memory 18 in step 36 of Figure 2 and 3; See also Col. 3, Lines 47-59).

Accordingly, Appellants respectfully submit that neither Drew nor Waldin, taken alone or in any combination, disclose or fairly suggest at least the above-noted features of Claim 1, and claims that depend therefrom. Independent Claims 22 and 41 recite features similar to those above-noted features of Claim 1 pointed out above which are not disclosed or suggested by the references. Thus, Claims 22 and 41, and claims that depend therefrom, are also neither disclosed nor suggested by the references, taken separately or in combination, for reasons similar to those set forth above regarding Claim 1. For at least those reasons set forth above, it is requested that the Board reverse the Examiner's rejection under 35 U.S.C. 103(a).

II. The Examiner has failed to establish a prima-facie case of obviousness under 35 U.S.C. §103(a) of Claims 67-70 as being unpatentable over U.S. Patent No.

6,928,555 to Drew in view of U.S. Patent No. 6,094,731 to Waldin, et al. and further in view of U.S. Patent No. 6,802,028 to Ruff et al.

A. Standard Regarding Obviousness and Claim Interpretation

The standard regarding obviousness and claim interpretation is set forth above in connection with a previous rejection under 35 U.S.C. 103.

B. The cited references of Drew (U.S. Patent No. 6,928,555) and Waldin et al. (U.S. Patent No. 6,094,731), and further in view of Ruff et al. (U. S. Patent No. 6,802,028) do not disclose or fairly suggest every element of Appellants' claimed invention as to have rendered Appellants' claimed invention obvious to one of ordinary skill in the art at the time the invention was made.

The Examiner rejects Claims 67-70 under 35 U.S.C. 103(a) as being obvious over Drew and Waldin, and further in view of U. S. Patent No. 6,802,028 to Ruff et al. (hereinafter "Ruff"). Appellants traverse this rejection as set forth below and respectfully request that the rejection be reversed.

The features of independent Claim 1 is discussed above with respect to Drew and Waldin. Claims 67-70 depend from independent Claim 1. Ruff is cited as support for disclosing features of the dependent Claims 67-70 which Drew and Waldin fail to disclose. Appellants note that Col. 7, Line 53-Col. 8, Line 34 of Ruff is apparently cited as support for teaching an antivirus unit included in a disk controller of a storage device, wherein the disk controller is a first disk controller of a plurality of disk controllers included in the storage device, the antivirus unit is a first antivirus unit of a plurality of antivirus units included in the storage device and each of said plurality of disk controllers includes a different one of said plurality of antivirus units.

Appellants respectfully submit that Ruff does not overcome the above-mentioned deficiencies of Drew and Waldin with respect to independent Claim 1. Appellants point out that nothing in Ruff corrects the deficiencies of Drew and Waldin, as pointed out above with respect to at least the above-noted features as recited in Claim 1 and as discussed above. Thus, combining Drew and Waldin with Ruff does not overcome the deficiencies of Drew and Waldin with respect to the foregoing features of Appellants' Claim 1, and claims that depend therefrom.

Claims 67-70 which depend from Claim 1 are neither disclosed nor suggested by the references for at least the same reasons as Claim 1. However, Appellants will point out some particular features of the dependent claims which are also neither disclosed nor suggested by the references.

Claim 67 recites, in relevant part, ***wherein the antivirus unit is included in a disk controller of the storage device.*** Claim 68 also recites, in relevant part, ***wherein the antivirus unit is included as software running on the disk controller.*** As support for disclosing features of Claims 67 and 68, Ruff at Col. 7, Line 53-Col. 8, Line 34 is cited. The foregoing citation refers to Ruff's Figure 3 which includes a virus detector 312 and virus remover 316 in a computer system 100. The detector 312 and remover 316 are separate from the controller 306. Page 6 of the Office Action states that "Ruff teaches an antivirus unit included in a disk controller". However, Appellant cannot locate where in Figure 3, or in the foregoing citation of Ruff, is there disclosure or suggestion of the foregoing features of Claims 67 and 68. In contrast, Ruff's Figure 3 illustrates the virus detector and remover as part of the computer system but not included in the controller.

Accordingly, Appellants respectfully submit that Drew, Waldin and Ruff, taken alone or in any combination, disclose or fairly suggest Claim 1, and claims that depend therefrom. For at least those reasons set forth above, it is requested that the Board reverse the Examiner's rejection under 35 U.S.C. 103(a).

CONCLUSION

For the reasons set forth herein, it is respectfully requested that the Board reverse all of the Examiner's rejections under 35 U.S.C. 102 and 103.

Respectfully submitted,
MUIRHEAD AND SATURNELLI, LLC

Date: March 19, 2009

Anne E. Saturnelli
Registration No. 41,290

Muirhead and Saturnelli, LLC
200 Friberg Parkway, Suite 1001
Westborough, MA 01581
T: (508) 898-8601
F: (508) 898-8602

CLAIMS APPENDIX

The claims on Appeal are as follows:

1. (Previously Presented) A computer implemented method of scanning a storage device for viruses, comprising:

determining, by the storage device, each track of the storage device that has been accessed for a write operation since a previous virus scan using information about tracks of the storage device without using file-based information, the file-based information including information about file structure, file system, and file type;

providing, to an antivirus unit by the storage device, information indicating which tracks of the storage device have been accessed for a write operation since the previous virus scan; and

scanning, by the antivirus unit using the information provided by the storage device, at least a portion of each track identified as having been accessed for a write operation since the previous virus scan for viruses, wherein scanning is performed without using the file-based information.

2. (Cancelled)

3. (Previously presented) A method, according to claim 1, wherein the portion corresponds to a sector of the storage device.

4. (Previously presented) A method, according to claim 1, wherein the portion corresponds to a subportion of the storage device.

5. (Previously presented) A method, according to claim 1, wherein said determining each track of the storage device that has been modified includes:

creating a table that is indexed according to each track and has entries indicating whether a corresponding track has been modified, the entries being cleared after a virus scan to indicate that no tracks have been modified; and

setting a specific one of the entries in response to a corresponding track of the storage device being subject to a write operation.

6. (Original) A method, according to claim 5, wherein creating the table includes copying an other table provided by the storage device.

7. (Original) A method, according to claim 5, wherein creating the table includes using an other table provided by the storage device.

Claims 8 - 21 (Cancelled).

22. (Previously Presented) A computer program product for scanning a storage device for viruses, the computer program product including a computer-readable medium with executable code stored thereon for:

determining, by the storage device, each track of the storage device that has been accessed for a write operation since a previous virus scan using information about tracks of the storage device without using file-based information, the file-based information including information about file structure, file system, and file type;

providing, to an antivirus unit by the storage device, information indicating which tracks of the storage device have been accessed for a write operation since the previous virus scan; and

scanning, by the antivirus unit using the information provided by the storage device, at least a portion of each track identified as having been accessed for a write operation since the previous virus scan for viruses, wherein scanning is performed without using the file-based information.

23. (Cancelled)

24. (Previously presented) A computer program product, according to claim 22, wherein the portion corresponds to a sector of the storage device.

25. (Previously presented) A computer program product according to claim 22, wherein the portion corresponds to a subportion of the storage device.

26. (Previously presented) A computer program product, according to claim 22, wherein said code for determining each track of the storage device that has been modified includes code for:

creating a table that is indexed according to each track and has entries indicating whether a corresponding track has been modified, the entries being cleared after a virus scan to indicate that no tracks have been modified; and

setting a specific one of the entries in response to a corresponding track of the storage device being subject to a write operation.

27. (Previously presented) A computer program product, according to claim 26, wherein said code for creating the table includes code for copying an other table provided by the storage device.

28. (Previously presented) A computer program product, according to claim 26, wherein said code for creating the table includes code for using an other table provided by the storage device.

Claims 29 - 40 (Cancelled).

41. (Previously Presented) An antivirus unit, comprising:

means for coupling to at least one storage device;

means for determining each track of the storage device that has been accessed for a write operation since a previous virus scan using information about tracks of the storage device without using file-based information, the file-based information including information about file structure, file system, and file type;

means for receiving, from the at least one storage device, information determined by the at least one storage device indicating which tracks of the at least one storage device have been accessed for a write operation since the previous virus scan; and

means for scanning, using the information provided by the storage device, at least a portion of each track identified as having been accessed for a write operation since the previous virus scan for viruses, wherein scanning is performed without using the file-based information.

42. (Canceled)

43. (Previously presented) An antivirus unit, according to claim 41, wherein the portion corresponds to a sector of the storage device.

44. (Previously presented) An antivirus unit, according to claim 41, wherein the portion corresponds to a subportion of the storage device.

45. (Previously presented) An antivirus unit, according to claim 41, further comprising:

a table that is indexed according to each track and has entries indicating whether a corresponding track has been modified, the entries being cleared after a virus scan to indicate that no tracks have been modified; and

means for setting a specific one of the entries in response to a corresponding track of the storage device being subject to a write operation.

46. (Original) An antrivirus scanning unit, according to claim 41, wherein said means for coupling includes means for coupling to only one storage device.

47. (Original) An antivirus unit, according to claim 41, wherein said means for coupling includes means for coupling to more than one storage device.

48. (Original) An antivirus unit, according to claim 41, further comprising:
means for coupling to at least one host.

49. (Original) An antivirus unit, according to claim 48, wherein said antivirus unit is interposed between said at least one storage device and said at least one host.

50. (Original) An antivirus unit, according to claim 48, wherein said antivirus unit is implemented as a process running on the at least one host.

51. (Original) An antivirus unit, according to claim 41, wherein said antivirus unit is implemented using stand alone hardware.

52. (Original) An antivirus unit, according to claim 41, wherein at least a portion of the antivirus unit is provided on at least some controllers for the at least one storage device.

Claims 53 - 62 (Cancelled).

63. (Previously presented) The method of Claim 1, wherein said storage device includes one or more sectors, and the method further comprising:

determining, for each sector of said storage device for a current virus scan, whether said each sector has been modified since a previous scan; and

for said current virus scan, scanning only those sectors determined to have been modified since said previous scan.

64. (Previously presented) The computer program product of Claim 22, wherein said storage device includes one or more sectors, and the computer-readable medium further comprising code stored thereon for:

determining, for each sector of said storage device for a current virus scan, whether said each sector has been modified since a previous scan; and

scanning only those sectors determined to have been modified since said previous scan.

65. (Previously presented) The antivirus unit of Claim 41, wherein said storage device includes one or more sectors, and the antivirus unit further comprising:

means for determining, for each sector of said storage device for a current virus scan, whether said each sector has been modified since a previous scan; and

means for scanning only those sectors determined to have been modified since said previous scan.

66. (Previously presented) The method of Claim 1, wherein the antivirus unit is included in the storage device.

67. (Previously presented) The method of Claim 66, wherein the antivirus unit is included in a disk controller of the storage device.

68. (Previously presented) The method of Claim 67, wherein the antivirus unit is included as software running on the disk controller.

69. (Previously presented) The method of Claim 67, wherein the antivirus unit is configured to use at least a portion of hardware that is separate from hardware of the disk controller.

70. (Previously presented) The method of Claim 67, wherein the disk controller is a first disk controller of a plurality of disk controllers included in the storage device, the antivirus unit is a first antivirus unit of a plurality of antivirus units included in the storage device, and each of said plurality of disk controllers includes a different one of said plurality of antivirus units.

71. (Previously presented) The antivirus unit of Claim 41, wherein said antivirus unit accessess data on the at least one storage device over a first connection and the information is provided on a second connection different from the first connection between said antivirus unit and the at least one storage device.

EVIDENCE APPENDIX

PAGE A1- Page 213 of Microsoft Computer Dictionary, Fifth Edition, which was cited in the list of references by the Examiner accompanying the Office Action dated October 5, 2005.

PAGE A2 - Page 223 from "UNIX INTERNALS, The New Frontiers" by Uresh Vahalia, which was submitted by Appellants in an Information Disclosure Statement filed December 14, 2005, and which was considered by the Examiner as indicated in a copy of the PTO-1449 form (as initialed by Examiner, the initialed PTO-1449 form accompanying the Office Action dated March 1, 2006).

RELATED PROCEEDINGS APPENDIX

None.

work users request files and make changes to them. To deal with the tasks of handling multiple—sometimes simultaneous—requests for files, a file server contains a processor and controlling software as well as a disk drive for storage. On local area networks, a file server is often a computer with a large hard disk that is dedicated only to the task of managing shared files. *Compare* disk server.

File Server for Macintosh *n.* An AppleTalk network integration service that allows Macintosh clients and personal computers clients to share files. *Also called:* MacFile. *See also* Print Server for Macintosh, Services for Macintosh.

file sharing *n.* The use of computer files on networks, wherein files are stored on a central computer or a server and are requested, reviewed, and modified by more than one individual. When a file is used with different programs or different computers, file sharing can require conversion to a mutually acceptable format. When a single file is shared by many people, access can be regulated through such means as password protection, security clearances, or file locking to prohibit changes to a file by more than one person at a time.

file size *n.* The length of a file, typically given in bytes. A computer file stored on disk actually has two file sizes, logical size and physical size. The logical file size corresponds to the file's actual size—the number of bytes it contains. The physical size refers to the amount of storage space allotted to the file on disk. Because space is set aside for a file in blocks of bytes, the last characters in the file might not completely fill the block (allocation unit) reserved for them. When this happens, the physical size is larger than the logical size of the file.

filespec *n. See* file specification (definition 1).

file specification *n.* **1.** The path to a file, from a disk drive through a chain of directory files to the file name that serves to locate a particular file. Abbreviated filespec. **2.** A file name containing wildcard characters that indicate which files among a group of similarly named files are requested. **3.** A document that describes the organization of data within a file.

file structure *n.* A description of a file or group of files that are to be treated together for some purpose. Such a description includes file layout and location for each file under consideration.

file system *n.* In an operating system, the overall structure in which files are named, stored, and organized. A file system consists of files, directories, or folders, and the information needed to locate and access these items. The term can also refer to the portion of an operating system that translates requests for file operations from an application program into low-level, sector-oriented tasks that can be understood by the drivers controlling the disk drives. *See also* driver.

file transfer *n.* The process of moving or transmitting a file from one location to another, as between two programs or over a network.

File Transfer Protocol *n. See* FTP[1] (definition 1).

file type *n.* A designation of the operational or structural characteristics of a file. A file's type is often identified in the file name, usually in the file name extension. *See also* file format.

fill[1] *n.* In computer graphics, the colored or patterned "paint" inside an enclosed figure, such as a circle. The portion of the shape that can be colored or patterned is the fill area. Drawing programs commonly offer tools for creating filled or nonfilled shapes; the user can specify color or pattern.

fill[2] *vb.* To add color or a pattern to the enclosed portion of a circle or other shape.

fill handle *n.* The small black square in the lower-right corner of a cell selection. When you point to the fill handle, the pointer changes to a black cross.

film at 11 *n.* A phrase sometimes seen in newsgroups. An allusion to a brief newsbreak on TV that refers to a top news story that will be covered in full on the 11 o'clock news, it is used sarcastically to ridicule a previous article's lack of timeliness or newsworthiness. *See also* newsgroup.

film recorder *n.* A device for capturing on 35-mm film the images displayed on a computer screen.

film ribbon *n. See* carbon ribbon.

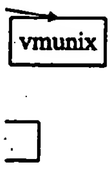
filter *n.* **1.** A program or set of features within a program that reads its standard or designated input, transforms the input in some desired way, and then writes the output to its standard or designated output destination. A database filter, for example, might flag information of a certain age. **2.** In communications and electronics, hardware or software that selectively passes certain elements of a signal and eliminates or minimizes others. A filter on a communications network, for example, must be designed to transmit a certain frequency but attenuate (dampen) frequencies above it (a lowpass filter), those below it (a highpass filter), or those above and below it (a bandpass filter). **3.** A pattern or mask through which data is passed to weed out specified items. For instance, a filter used in e-mail or in retrieving newsgroup messages can allow users to filter

.e-structured name space
.e leaf nodes.[1] A directory
ı it. Each file or directory
acter. The file system may
". Filenames only need to
ctories have a file called
pathname. The pathname
the node, separated by '/'
ıt different pathnames—
he name of the root direc-

:h process, maintained as
ive pathnames, which are
ıe components: the first is
the parent directory. The
ory itself. In Figure 8-1, a
ither as /usr/lib *(absolute*
ıt directory by making the

to that file. Any file may
Thus a file is not bound to
ribute of the file. The file
nks are equal in all ways
l through any of its links,
systems also provide an-

format. Since application
y, the POSIX.1 standard



(ignoring the ".." entries that
: simpler and just as adequate.

specifies the following standard library routines to operate on directories:

```
dirp = opendir (const char *filename);
direntp = readdir (dirp);
rewinddir (dirp);
status = closedir (dirp);
```

These routines were first introduced in 4BSD and are now supported by SVR4 and most commercial variants. When the user calls *opendir,* the library associates a directory stream with it and returns a stream handle to the user. The stream object maintains an offset to the next entry to be read. Each *readdir* call returns a single directory entry and advances the offset. The entries are returned in file-system-independent format, defined by the following structure:

```
struct dirent {
    ino_t d_ino;                  /* inode number (see Section 8.2.2) */
    char  d_name[NAME_MAX + 1];   /* null-terminated filename */
};
```

The value of NAME_MAX depends on the file system type. SVR4 also provides a *getdents* system call to read directory entries in file-system-independent format. The format of entries returned by *getdents* is different from that of `struct dirent`. Hence users should use the more portable POSIX functions wherever possible.

8.2.2 File Attributes

Apart from the file name, the file system maintains a set of attributes for each file. These attributes are stored not in the directory entry, but in an on-disk structure typically called an *inode*. The word *inode* is derived from *index node*. The exact format and contents of the inode are not the same for all file systems. The *stat* and *fstat* system calls return the file's attributes in a filesystem-independent format. The commonly supported attributes of a file include the following:

- File type — Besides regular files, UNIX® recognizes several special types of files including directories, FIFOs (first-in, first-out files), symbolic links, and special files that represent block or character devices.
- Number of hard links to the file.
- File size in bytes.
- Device ID — Identifies the device on which the file is located.
- Inode number — There is a single inode associated with each file or directory regardless of how many links it has. Each inode on a given disk partition (logical disk, see Section 8.3.1) has a unique *inode number*. Hence a file is uniquely identified by specifying its device ID and inode number. These identifiers are not stored in the inode itself. The device ID is a property of the file system—all files of a single file system have the same device ID. Directory entries store the inode number along with the filename.
- User and group IDs of the owner of the file.

BEST AVAILABLE COPY